

## Tug of War Game – Arduino Shield Assignment

Create a tug-of-war game as we did in class, but with a few changes. To facilitate some fancier programming later, we need to adjust the pinouts of our game as well as add a speaker (“piezo buzzer”). Here are your new pinouts:

- Button 1            Pin 2
- Button 2            Pin 3
- LEDs                Pins 4-12 inclusive
- Speaker             Pin A0 (yes, we can use the analog pins as digital pins! Input AND Output)

Here are the components you should be selecting within the schematics editor:

- Resistor:            R
- LED:                 LED
- Button:             SW\_PUSH
- Speaker:            Speaker
- Arduino Shield: Arduino\_Uno\_R3

Here are the footprints you should be PCB editor:

- Resistor:            Resistor\_THT:R\_Axial\_DIN0207\_L6.3mm\_D2.5mm\_P7.62mm\_Horizontal
- LED:                 LED\_THT:LED\_D5.0mm
- Button:             Button\_Switch\_THT:SW\_PUSH\_6mm\_H4.3mm
- Speaker:            Buzzer\_Beeper:MagneticBuzzer\_StarMicronics\_HMB-06\_HMB-12
- Arduino Shield: Module:Arduino\_UNO\_R3

### 1. Circuit Design and PCB design [ DUE: By the end of class - Friday February 5<sup>th</sup>, 2021 ]

- Using KiCAD, construct a schematic diagram that represents your functional prototype of the above game.
- Associate parts to your PCB footprints
- Layout your PCB.
  - **Be sure to set the following tolerances under Design Rules before you begin (File → Board Setup → Net Classes)**
  - Default clearance of 0.4mm
  - Default track width of 0.5mm
  - NOTE: This will make it harder to wire, but is necessary when making our own boards
- Remember that all parts must be placed within the footprint for the Arduino Shield
- Add some text (on the copper layer) that says what class you are in (TEJ4M), your name, and the year.
- Consider printing out the PCB on paper to verify that you have a comfortable layout for the game.
- Once you have a confirmed layout and wiring, your circuit will be etched onto a copper board and cut to size.

### 2. Soldering and final PCB assembly [ DUE: To be announced based on timing ]

- You will be provided with your etched board with holes already drilled.
- Solder your final printed circuit board and test on the Arduino.
- If not all connections work, check soldering, check the traces for breaks and finally check parts

### 3. Simple Tug-of-War Programming [ DUE: To be announced based on timing ]

- Create a basic game using this shield and your Arduino. Feel free to test the code using a simulation on TinkerCAD that has the same wiring via a breadboard.

## Marking Rubric

<b>Expectation</b>	<b>R</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>
<b><i>KiCAD</i></b>	<p>KiCAD schematic is clear and difficult to follow / does not have a correct function</p> <p>KiCAD PCB layout is missing key components / connections</p>	<p>KiCAD schematic is functional and not missing any connections</p> <p>KiCAD PCB layout has all connections wired.</p>	<p>KiCAD schematic is functional and connections are clear</p> <p>KiCAD PCB layout has functional layout</p>	<p>KiCAD schematic is mostly clear and well laid out</p> <p>KiCAD PCB layout has functional layout, has some symmetry and elegance.</p>	<p>KiCAD schematic is clear and very well laid out</p> <p>KiCAD PCB layout has a good layout, has much symmetry and elegance.</p>
<b><i>Soldering</i></b>	<p>Large sections of soldering not done.</p>	<p>Some soldering is missing.</p>	<p>Soldering of all components has been attempted.</p>	<p>All soldering has been done on all components, at least functionally.</p>	<p>All soldering has been done on all components with good solder joints.</p>
<b><i>Arduino Code</i></b>	<p>Code does not properly use LEDs / buttons</p> <p>Tug-of-War game is not close to playable</p>	<p>Code makes some use of LEDs / buttons but has problems</p> <p>Tug-of-War game is obviously the goal, but does not work fully</p>	<p>Code uses all buttons &amp; LEDs but has a few glitches / delays</p> <p>Tug-of-War game is nearly functional with only small glitches</p>	<p>Code is able to use all buttons and LEDs</p> <p>Tug-of-War game is functional</p>	<p>Code is able to use all buttons, LEDs and uses the speaker for game feedback.</p> <p>Tug-of-War game is fully functional and includes some kind of notification that a player has won a point / won the game.</p>